TDIG Engineering Manual

Table of Contents:

Change History:

Version 3_0 : October 1, 2007
- Formatting changes
- Settings for Aux data path readout, including J17 settings and TDC configuration.
- J14 description (TDC clock monitor connector)
- LED indicator description
- Appendix A: TDIG FPGA configuration registers

Block diagram

Quick Start

JU1 – CAN bus termination. Short 1-2 enables termination

JU2 – General purpose MCU jumper inputs.
- Short 1-2 to select external oscillator (40 Mhz clock from TCPU)
- Open 1-2 to select on-board oscillator

- 3-4 not used
- 5-6 msb of jtag tdc select (short = 1) (only used with JTAG TDC I/O)
- 7-8 lsb of jtag tdc select (short = 1) (only used with JTAG TDC I/O)

JU3 – Clock and J9 routing
- 1-2 short allows local clock from J9
- **3-4 short allows J9 as I/O with FPGA /pin AA20**
- 5-6 short allows local clock from on-board oscillator
- 7-8 short grounds local clock input

Programming header placement: 4 connectors in upper left hand side of board.
J8 in corner
J6 next below
J7 next below
J10 next below

J8 : Byteblaster programming header for FPGA programming (configuration eeprom #1)
J6: Byteblaster JTAG header for FPGA debug
J7 : Byteblaster JTAG header for CPLD programming
J10 : ICD2 header for MCU programming

J11 : MCU reset select
Short 1-2 to allow ICD2 programming
Short 2-3 to allow hardware reset from SW3 or TCPU reset signal
Default : Leave jumper in **short 1-2** position

J12: CAN bus header.
J12/pin1 = can_N
J12/pin2 = gnd
J12/pin3 = can_H

J14: Board clock out. 40 Mhz, PECL levels. Clock that goes to FPGA, HPTDCs, after all clock switching.

J17: Aux data path token routing – jumper 2-3 if board is farthest from TCPU; otherwise jumper 1-2.

SW4: board position switch. Selects board position in tray, position 0 to 7.

Configuration reference

Jumper settings:

JU1

JU2

PINS	OPEN	SHORT
1-2	Selects internal clock (see JU3 settings for source of internal clock)	Selects external clock from upstream ribbon cable
3-4	Not used	Not used
5-6	MSB of JTAG select = 1	MSB of JTAG select = 0
7-8	LSB of JTAG select = 1	LSB of JTAG select = 0

JTAG SELECT		

CONFIG_1.1	CONFIG_1.0	
MSB	LSB	SELECTION
0	0	No active TDC
0	1	TDC #3 (U4)
1	0	TDC #2 (U3)
1	1	TDC #1 (U2)

The value read by the MCU during a polling routine from JU2 (5-6: 7-8) is written as the JTAG select field to TDIG fpga configuration register CONFIG_1(1:0) as shown in the table, and the 2 bits are decoded to enable TDC selection as shown.

JU3

Power Sequencing

Power on sequencing is controlled by the MCU. The MCU enables dedicated regulators for the TDC I/O and core voltages. Upon enable from MCU ("ENABLE_TDC_POWER" valid), the TDC I/O regulator is enabled. That regulator's output voltage then enables the TDC core regulator.

The MCU (Microchip PIC24HJ64GP506) runs entirely from 3.3 Volts at 40 Mhz. After it's internally controlled Power-on reset, the MCU can control and monitor the remaining voltages in the system.

Resets

Power on reset occurs internally to the MCU. After initialization, the MCU resets the TDCs, initiates PLD configuration, resets the PLD and configures the TDCs. TDC reset is independent of PLD reset.

Clock Distribution

The TDIG system clock, 'CLK_40MHZ', drives the MCU, PLD and the TDCs. This clock has 2 sources: an on-board oscillator and an external clock from TCPU.

The upstream input clock is buffered and split with U?, with one leg going directly to the downstream connector. With this arrangement, a failrue in the clock switching control will not affect operation of downstream boards.

UCLK_40MHZ is from the upstream connector. This signal is fanned out by the 2:1 in / 5-out PECL part. The attenuated (zero jitter PECL to LVDS conversion) clocks go to the TDCs, plus a conversion to ttl and a second fanout to MCU and PLD.

The local osc. is selected by the MCU and can be disabled by the MCU when not in use. Only one of the 2 local osc. footprints will be stuffed. There's also a test clock input that can be jumpered in from an RF connector.

The MCU resets to local osc. and switches to external osc. under CAN Bus command for normal system operation. If the ext. clock fails in operation, the MCU will automatically switch to it's own internal oscillator and execute a trap. The trap code will switch back to the local oscillator. So an external clock failure does not hang the micro.

The local oscillator is also routed to the PLD for use as an asynchronous source for testing, including the generation of the tray test pulse that is sent to TINO.

Aux data Readout

To use the AUX readout path, you must:

1. Set HPTDC configuration bit #44 “SELECT BYPASS INPUTS” to “1” in all TDCs in the readout chain.
2. Set TCPU FPGA configuration register bit CONFIG_14.7 = “1”
3. Unless power is cycled, a reset to the TCPU state machines, and possibly the HPTDCs, will probably be necessary when switching between modes (toggle CONFIG_1.0 HI then LO (“002 5 0E 01 00 01 00”).
4. Set jumper J17 properly on each TDIG in the readout chain for Aux data path token routing – Jumper 2-3 if board is farthest from TCPU; otherwise jumper 1-2.

Reading data with JTAG

Set up data to TDCs:

Mount front end test fixture to TDIG
Set up data input to front end test fixture

OR

Enable test data input to chan1 on TDCs using FPGA configuration registers

Attach Byteblaster to JTAG readout cable and test header

Select or edit the “tdig.jam” file so the the config_basic command gives the appropriate TDC configuration. Set the TDC ID field. Edit “data_file” command to give the desired number of words in the output file.

Open a DOS window and issue the following commands:

```
jam_tdc -aconfig_basic tdig.jam
jam_tdc -alock tdig.jam
```

TURN OFF THE DATA SOURCE

```
jam_tdc -areset tdig.jam

jam_tdc -astatus tdig.jam
```

CHECK THAT THE READOUT FIFO IS EMPTY
TURN ON THE DATA SOURCE

```
jam_tdc -astatus tdig.jam
```

CHECK THAT THE READOUT FIFO IS FULL

```
jam_tdc -adata tdig.jam
```

-- reads single data words

ONE WORD WILL BE READ – CHECK FOR CORRECT FORMAT AND TDC ID.

```
jam_tdc -adata_file -otesta.dat tdig.jam
```

-- reads data to file, edit jam code to control # of words.

Upstream / Downstream signals over ribbon cable

Due to placement of the upstream ribbon cable connector on the top side of the circuit card, the ribbon cables invert the polarity of their differential signals.

All signals discussed in this section are labelled as

- OUT (from TDIG to upstream) or
- IN (from upstream to TDIG)

The re-inversions take place on the UPSTREAM side of TDIG only.

Some of these signals are re-inverted by swapping pin polarities at the upstream / downstream connectors:

- UCLK_40MHZ (IN)
- CAN (BIDIRECTIONAL)

Some signals are re-inverted by swapping pin polarities at the upstream receiver device (U22):

- TRIGGER (IN)
- CLK_10MHZ (IN)
- BUNCH_RST (IN)

Some of these signals should be re-inverted by swapping pin polarities at the receiver/driver devices, but are NOT so far (TDIG-E circuit card). These will be swapped on future versions of the board. These signals are used only in the auxilliary serial readout data path:

H3_SER_OUT
H3_TOKEN_OUT
H3_STROBE_OUT

--

The signals below are re-inverted at the UPSTREAM input/output of the TDIG FPGA. No inversion takes place on the DOWNSTREAM side.

UDAISY_DATA (OUT)
UDAISY_TOK_OUT (OUT)
USTATUS0 (OUT)
USTATUS1 (OUT)
UDAISY_CLK (OUT)
UDAISY_TOK_IN (IN)
MCU_RESET?? (IN)
FLEX_RESET_IN (IN)
USPARE_IN1 (IN)
UCONFIG_IN (IN)
UMULT[3:0} (OUT)

Appendix A (attached) : “TDIG FPGA – MCU IF registers ver 1.xls” Definition of R/W registers in FPGA that provide configuration / status communication with MCU.

LED Indicators on TDIG-E and F

For TDIG- MCU firmware version 11D or later (Release 5 or later) any or all LEDs may illuminate briefly during MCU firmware initialization and start-up. LEDs D9 and D11 are connected to specific hardware signals as described below. LEDs D0 through D7 are controlled through the MCU I^2C bus I/O expander port. Writing a "0" bit to the port turns the corresponding LED on.

Once the MCU program has initialized LEDs are used as follows ***unless*** overridden by a CANBus message (write_led, 0x1?4, length 2, payload '0x0A' <byte>).

Reading from top to bottom:

D0 (green) – This LED reflects the current state of jumper position JU2 pins 7-8. Jumper installed results in LED-ON. This jumper position is used in conjunction with JU2 pins 5-6 to select an HPTDC for manual JTAG access. Jumper installed = Bit 2^0 selected.
D1 (green) – This LED reflects the current state of jumper position JU2 pins 5-6. Jumper installed results in LED-ON.. This jumper position is used in conjunction with JU2 pins 7-8 to select an HPTDC for manual JTAG access. Jumper installed = Bit 2^1 selected.
D2 (green) – This LED reflects the current state of jumper position JU2 pins 3-4. Jumper installed results in LED-ON. This jumper position is not currently used.
D3 (green) – This LED reflects the current state of jumper position JU2 pins 1-2. Jumper installed results in LED-ON and selects "Tray Clock" from connector J5 as the board clock source.
D4 (green) – This LED lit indicates an error condition. It will be illuminated if the configuration (647-bit setup string) read-back from one or more of the HPTDC chips differs from what was programmed.

D5 (green) – This LED is not used at present.
D6 (green) – This LED will be lit when the HPTDCs have been configured.
D7 (amber) – This LED is illuminated when the MCU requests the HPTDC power regulator "ON" condition.
D9 (green) – This indicator is hard-wired to the MCU "G"-Register bit 15 (MCU_TEST) and test-point TP2. A low-level at TP2 corresponds to LED-on. There is a 50% duty-cycle square wave generated by the MCU to this testpoint when the MCU program is running in the "idle" loop after all initializations are done. The LED will appear slightly "dim" relative to the others when the idle loop is functioning.
D11 (green) – This indicator is hard-wired to the FPGA. If the FPGA has not been programmed it will be off. After the FPGA initializes, It will normally be lit. Pressing SW3 will cause the LED to turn off.

The normal idling condition of a TDIG board will be:
D11, D9, D7, D6, D3 = ON; the remaining LEDs = OFF

Appendix A: TDIG FPGA configuration registers

	Byte	**Register name**	**Bit**	**Description**	**Notes**
	Address		**Address**	(Active High unless indicated)	
MCU writes to:			(7 is msb)		
	0	CONFIG_0		Configuration 0	
			0	Select test input to MCU FIFO	DEFAULT = 0
			1	Select this board as first board in readout chain: Token to first TDC comes from upstream (or MCU in test mode) rather than from downstream TDIG.	DEFAULT = 0
			2	Select test mode for serial readout: FPGA uses Strobe 0 as signal to issue token and do serial readout from TDCs	DEFAULT = 0
			3	Select test mode for TDC data: FPGA uses STROBE_1 as signal to issue one pulse on Channel 1 to each TDC.	DEFAULT = 0
			4	Select test mode for TDC trigger: FPGA uses STROBE_2 as signal to issue one trigger pulse to each TDC.	DEFAULT = 0
			5	Select test mode for Bunch Reset	DEFAULT = 0 (bunch reset from upstream)
			6	Select test mode for Event Reset	DEFAULT = 0 (event reset from upstream)
			7		
	1	CONFIG_1			
			1:0	Select TDC for JTAG communication (msb) "JTAG_MODE" msb =JU2 / 5,6 lsb = JU2 / 7,8 msb | lsb 00 : no TDC selected 01 : TDC1 (U2) selected 10 : TDC2 (U3) selected 11 : TDC3 (U4) selected	For user communication w/ TDCs using the Byteblaster connected to the Test Header, MCU reads jumper settings to select TDC. Default board configuration should have these jumpers open (unstuffed), so that MCU will write 00.

			2	Select MCU as source for TDC configuration. "JTAG_SEL" If not selected, then JTAG input from test header is selected.	DEFAULT = 0
			4:3	Not currently implemented. TDC readout selection: table index = bits 765 000 : 3 TDCs 001 : TDC U2 010 : TDC U3 100 : TDC U4	DEFAULT = 000
			5		
			6		
			7		
	2	CONFIG_2			
			0	TDC hardware reset	Active HI; DEFAULT = 0
			1	Bunch reset test mode	0 = shift register source, 1 = mcu strobe 7, default = 0
			2		
			3		
			4		
			5		
			6		
			7		
	3	CONFIG_3			
	4	STROBE_4	N/A	Generate test token. Upon receipt of this signal, the serial readout controller will issue token and do serial readout from TDCs. Enabled by CONFIG_0.2.	
	5	STROBE_5	N/A	Generate test data. FPGA will generate 1 test pulse for each TDC for each STROBE_1 input. Enabled by CONFIG_0.3.	
	6	STROBE_6	N/A	Generate test trigger. FPGA will generate 1 trigger pulse for each TDC for each STROBE_2 input. Enabled by CONFIG_0.4.	
	7	STROBE_7	N/A	Generate test bunch reset. FPGA will generate 1 bunch reset pulse for each TDC for each STROBE_3 input. Enabled by CONFIG_0.5	

	8	STROBE_8	N/A	Generate test event reset. FPGA will generate 1 event reset pulse for each TDC for each STROBE_3 input. Enabled by CONFIG_0.5	
	9	STROBE_9	N/A	Reset readout. Resets local serial readout state machine.	
	10	STROBE_10	N/A	Reset MCU FIFO	
	11	STROBE_11	N/A	Clocks test data counter and mcu fifo input	
	12	CONFIG12	2:00	Position switch (2:0) right justified i.e. 3 lsbs	
	13				
	14	CONFIG14	0	SELECT LVDS TEST	
	15				

	Byte	**Register name**	**Bit**	**Description**	**Notes**
	Address		**Address**	(Active High unless indicated)	
MCU reads from:					
	0	Configuration 0 readback			
	1	Configuration 1 readback			
	2	Configuration 2 readback			
	3	Configuration 3 readback			
	4				
	5				
	6				
	7	Version ID		Constant value can be used to ID FPGA code version	
	8	Status 0			
			0		
			1		
			2		
			3		
			4		
			5		
			6		
			7		
	9	Status 1			
			0		
			1		
			2		

			3		
			4		
			5		
			6		
			7		
	10				
	11	FIFO (7:0)		LS byte	
	12	FIFO (15:8)			
	13	FIFO (23:16)			
	14	FIFO (31: 24)		MS byte Reads should be from LSbyte (read first) to MSbyte (read last). Final read from MS byte generates read_clock_enable signal to FIFO.	
	15	FIFO status			
			4:0	MCU FIFO OCCUPANCY (# of data words in FIFO)	
			5	MCU FIFO PARITY	
			6	MCU FIFO FULL	
			7	MCU FIFO EMPTY	